Victoria University of Bangladesh
Dept. of CSE
Program B.Sc in CSIT
Semester:- Spring' 2023
Course title:- Discrete Mathematics
Course code:- MAT 135

Name:- Nusrat Jahan Tanshee
ID:- 2520200011
Batch:- 20

Final assesment

1

Ans to the Q no- 1 -a

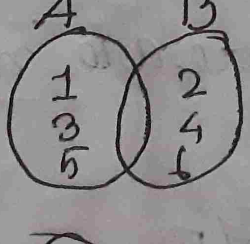
Universal set:- A universal set is a set which contains all the elements or ~~subjects~~ objects of other sets, including its own elements. It is usually denoted by the symbol 'U'.

Example:- $A = \{2, 4, 6, 8, 10\}$ and $B = \{1, 3, 5, 7, 9\}$

$$U = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$$

Empty set:- A set which does not contain any element is called the empty set ~~ot~~ or the null set or the void set. It is usually denoted by the symbol "$\emptyset$" or "ϕ" or "$\{\ \}$".

Example:- Venn diagram representation of an empty set.



2

Disjoint set:- When the intersection of two sets is a null or empty set, then they are called disjoint sets. Hence, if A and B are two disjoint sets, then;

$A \cap B = \phi$

Example:- $A = \{2, 3\}$ and $B = \{4, 5\}$ are disjoint.

$C = \{3, 4, 5\}$ and $\{3, 6, 7\}$ are not disjoint as both the sets C are having 3 as an common elements.

Ans to the Q no-1-b

In counting the element of $A \cup B$, first count those that are in A. There are $n(A)$ of these. The only other element of $A \cup B$ are those that are in B but not in A. But since A and B are disjoint no element of B in in A, so there are $n(B)$ element that are in B but not in A.

Therefore, $n(A \cup B) = n(A) + n(B)$.

For any sets A and B, the set A is the disjoint union of A \ B and A ∩ B. Thus Lemma rules gives us the useful result.

Ans to the Q no – 2(a)

Breadth-first search (BFS) is an algorithm for searching a tree data structure for a node that satisfies a given property. It starts at the tree root and explores all the nodes at the present depth prior to moving on to the nodes at the next depth level. Extra memory, usually a queue is needed to keep track of the child notes that were encountered but not yet explored.

Algorithm 1.1:- Breadth-first-search-

This algorithm executes a breadth-first search on a directed graph G beginning with a starting vertex A.

Step 1:- Initialize all verticles to the ready state.
(Status=1)

Step 2:- Put the starting vertex A in Queue and change the status of A to the waiting state.
(Status = 2)

Step 3:- Repeat steps 4 and 5 until Queue is ~~ep~~ ~~emp~~ empty.

Step 4:- Remove the vertex N of Queue. Process N and set Status(N) = 3, the processed state.

Step 5:- Examine each neighbor J of N.

a) If status (J) = 1 (ready state), add J to the rear of Queue and ~~rest~~ reset Status(J) = 2 (waiting state).

b) If status (J) = 2 (waiting state) or Status (J) = 3 (Processed state), ignore the vertex J.

[End of step 3 loop]

Step 6: Exit.

Algorithm 1.1 will process only those vertices which are reachable from a starting vertex A. Suppose one wants to process all the vertices in the graph G. Then the algorithm must be modified so that it begins again with another vertex that is still in the ready

State (State-1). This new vertex, say, B, can be obtained by traversing through the list of vertices.

Ans to the Q no - 2(b)

Welch-Powell Algorithm:- It is a graph colouring algorithm. In graph theory, vertex colouring is a way of labelling each individual vertex such that no two adjacent vertex have same colour. But we need to find out the number of colours. We need to satisfy the given condition. It is not desirable to have a large variety of colours or labels. So, we have an algorithm called welch-powell algorithm that gives the minimum colours we need. This algorithm is also used to find the chromatic number of a graph. This is an iterative greedy approach.

Algorithm 1.2: Welch - Powell -

The input is a graph G.

Step 1: Order the vertices of G according to decreasing degrees.

Step 2: Assign the first colour C_1 to the first vertex and then in sequential order assign C_1 to each vertex which is not adjacent to a previous vertex which was assigned C1.

Step 3: Repeat step 2 with a second colour C_2 and and the subsequence of noncolored vertices.

Step 4:- Repeat step 3 with a third colour C_3, then a fourth color C_4 and so on untill all vertces are colored.

Step 5:- Exit.

By starting with the highest degree, we make sure that the vertex with the highest number of conflicts can be taken care of as possible.

Ans to the Q no- 3(a)

Let, M = {Students taking mathematics} and
C = {students tanking chemistry}.

Since the space is equiprobable,

$P(M) = \frac{40}{100} = \frac{4}{10}$, $P(C) = \frac{30}{100} = \frac{3}{10}$,

$P(M \text{ and } C) = P(M \cap C) = \frac{20}{100} = \frac{1}{5}$

Thus, by the addition principle:—

For any events A and B

$P(A \cup B) = P(A) + P(B) - P(A \cap B)$

$\therefore P = P(m \text{ or } c) = P(M \cup C) = P(M) + P(C) - P(M \cap C)$

$= \frac{4}{10} + \frac{3}{10} - \frac{1}{5}$

$= \frac{1}{2}$

Ans to the Qs no - 3(b)

Let, A = {the card in a spade} and B = {the card is a face card}

We compute P(A), P(B) and P(A∩B). Since we have an equiprobable space,

$P(A) = \frac{\text{number of spades}}{\text{number of cards}} = \frac{13}{52} = \frac{1}{4}$

$P(B) = \frac{\text{number of face cards}}{\text{number of cards}} = \frac{12}{52} = \frac{3}{12}$

$P(A \cap B) = \frac{\text{number of spade face cards}}{\text{number of cards}} = \frac{3}{52}$

Theorems on finite probablity spaces:-

The following theorem follows directly from the fact that the probability of an event is the sum of the probabilities of its points.

Theorem 1.1: The probability function P defined on the class of all events in a finite probability space has the following properties:

$\boxed{P_1}$ For every event A, $0 \leq P(A) \leq 1$

$\boxed{P_2}$ $P(S) = 1$

$\boxed{P_3}$ If events A and B are mutually exclusive, then $P(A \cup B) = P(A) + P(B)$

The next theorem formalizes over intuition that if p is the probability that one event E occurs, then 1-p is the probability that E does not occur. (That is if we hit a target p=1/3 of the times, then we miss the target 1-p = 2/3 of the times)

Theorem 1.2: Let A be any event. Then $P(A^c) = 1 - P(A)$

The following theorem (Proved in problem) follows directly from theorem 1.1.

Theorem 1.3: Consider the empty set ϕ and any events A and B then:

(i) $P(\phi)=0$

(ii) $P(A \setminus B) = P(A) - P(A \cap B)$

(iii) If $A \subseteq B$, then $P(A) \leq P(B)$

Observe that property $[P_3]$ in theorem 1.1 gives the probability of the union of events in the case that the events are disjoint. The general formula is called the addition principle specifically.

Theorem 1.4: For any events A and B

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

Ans to the Q no-4(a)

We seek the number of permutations of 7 objects of which 3 are alike (the three E's), and 2 are alike (the two N's)

By theorem,

$$P(n; n_1, n_2, \ldots, n_r) = \frac{n!}{n_1!\, n_2! \ldots n_r!}$$

$$\therefore m = P(7,3,2) = \frac{7!}{3!\,2!} = \frac{7.6.5.4.3.2.1}{3.2.1.2.1} = 420$$

Ordered Samples:-

Many problems are concerned with choosing an element from a set S, say with n elements. When we choose one element after another, say, r, times we call the choice an ordered sample of size r. We consider two cases:-

(1) Sampling with replacement:- Here the element is replaced in the set S before the next element is chosen. Thus, each time there are n ways to choose an element (repititions are allowed). The product rule tells us that the number of such samples is:

$$n.n.n \ldots n.n \ldots (r \text{ factors}) = n^r$$

(2) Sampling without replacement:- Here the element isn't replaced ~~at~~ in the set S before the next element is chosen. Thus, there is no repitition in the ordered sample. Such a sample is simply an r-premutation. Thus the number of such sample is:

$$P(n, r) = n(n-1)\cdot(n-2)\ldots(n-r+1) = \frac{n!}{(n-r)!}$$